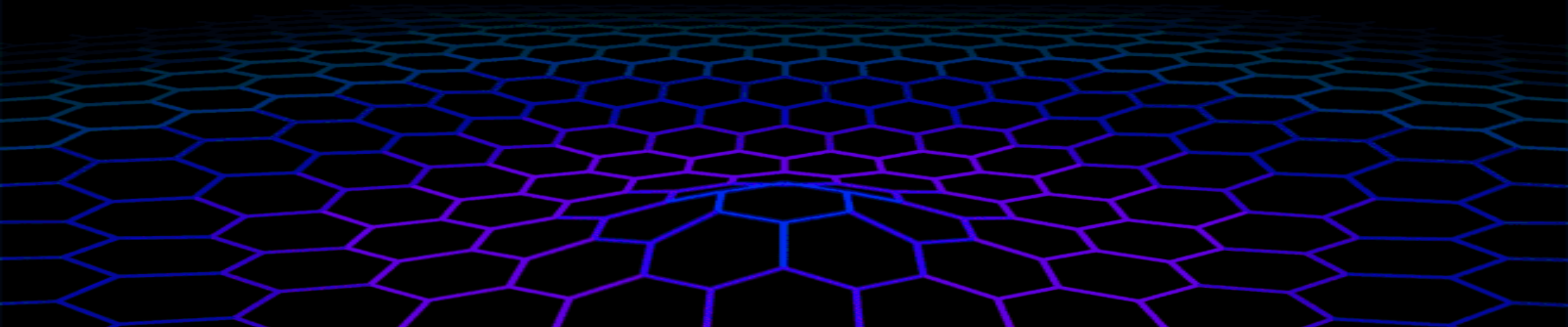


Banco de Dados I






SQL

Tabela PAI



A **coluna 1** é a **chave primária**.
Todos valores são **únicos**.

<u>Coluna 1</u> 	Coluna 2
A	X
B	Y
C	X
D	Y

INTEGRIDADE REFERENCIAL

Tabela FILHO

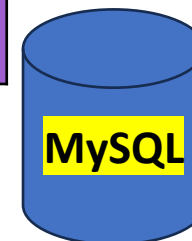
A **coluna 2** é **chave estrangeira**.
Todos os valores devem
pertencer à coluna de referência
da tabela pai.

<u>Coluna 1</u> 	Coluna 2 
1	A
2	NULL
3	C
4	A

Quero inserir
O dado '**G**'
na **Coluna 2**
da tabela Filho



5	G
---	---



Erro: Não posso permitir
a entrada desta linha. '**G**'
não está na coluna 1 da
tabela pai.

Chave Estrangeira no MySQL

```
CREATE TABLE cidades (  
  id    INT AUTO_INCREMENT,  
  nome  VARCHAR(250) NOT NULL,  
  PRIMARY KEY (id));
```

```
CREATE TABLE clientes (  
  id          INT NOT NULL AUTO_INCREMENT,  
  nome        VARCHAR(100) NOT NULL,  
  cidade_id  INT DEFAULT 15,  
  PRIMARY KEY (id));
```

```
ALTER TABLE clientes(  
  ADD CONSTRAINT FK_Cidade  
  FOREIGN KEY (cidade_id)  
    REFERENCES cidades(id)  
    ON DELETE SET NULL  
    ON UPDATE CASCADE);
```

Opções de Chave Estrangeira no MySQL

CASCADE: Permite excluir ou atualizar os registros relacionados presentes na tabela filha automaticamente, quando um registro da tabela pai for atualizado (ON UPDATE) ou excluído (ON DELETE). É a opção mais comum aplicada.

RESTRICT: Impede que ocorra a exclusão ou a atualização de um registro da tabela pai, caso ainda haja registros na tabela filha. Uma exceção de violação de chave estrangeira é retornada. A verificação de integridade referencial é realizada antes de tentar executar a instrução UPDATE ou DELETE

SET NULL: Esta opção é usada para definir com o valor NULL o campo na tabela filha quando um registro da tabela pai for atualizado ou excluído.

NO ACTION: Essa opção equivale à opção RESTRICT, porém a verificação de integridade referencial é executada após a tentativa de alterar a tabela. É a opção padrão, aplicada caso nenhuma das opções seja definida na criação da chave estrangeira.

Operadores Aritméticos:

São responsáveis pela execução de operações matemáticas simples:

+	Adição
-	Subtração
*	Multiplicação
/	Divisão

Operadores Relacionais:

- São utilizados quando precisamos fazer comparações entre dois valores:

>	Maior que
<	Menor que
=	Igual a
<>	Diferente de
>=	Maior ou igual a
<=	Menor ou igual a

Operadores lógicos:

AND (&&)

- O operador lógico **AND**, ou **E**, deve ser usado em uma pesquisa que se deseja entrar dois valores.
- O **AND**, verifica ambas as cláusulas da comparação, e só retorna algum valor se as duas tiverem uma resposta verdadeira.
- Exemplo:

```
SELECT * FROM teste WHERE (nome = 'Paulo Roberto') AND (telefone = '4834');
```

Esta pesquisa mostrara todos os registros que contém no campo nome o conteúdo '**Paulo Roberto**', **E (AND)** no campo telefone, o conteúdo '**4834**'.

Operadores lógicos:

OR (||)

- O operador lógico **OR**, ou **OU**, deve ser usado em uma pesquisa que se deseja entrar dois valores.
- O **OR**, verifica ambas as cláusulas da comparação, e retorna valores se qualquer um dos membros obtiver resultado.

Exemplo:

```
SELECT * FROM teste WHERE (nome = 'Paulo Roberto') OR (telefone = '4834');
```

Esta pesquisa fará com que todos os resultados que contenham o conteúdo '**Paulo Roberto**' no campo nome, **OU (OR)** telefone '**4834**' sejam exibidos na tela.

Operadores lógicos:

NOT (!)

- O operador lógico **NOT**, ou **NÃO**, realiza uma pesquisa, excluindo valores determinados do resultado.

Exemplo:

```
SELECT * FROM teste WHERE (nome != 'Paulo Roberto');
```

Esta pesquisa listará todos os registros da base de dados teste, **NÃO (NOT)** mostrando aqueles que possuem '**Paulo Roberto**' como conteúdo do campo nome.

Ordenação

ORDER BY

- **ORDER BY**, ou **ORDENAR POR**, simplesmente lista os registros, colocando-os em ordem de acordo com o campo solicitado.

```
SELECT * FROM teste WHERE (nome = 'Paulo') ORDER BY telefone;
```

O resultado desta busca resultara em todos os registros contendo 'Paulo' no campo nome, e a listagem será organizada de acordo com a ordem do telefone.

ORDER BY

ASC e **DESC** especificam o tipo de classificação e são, respectivamente, abreviações das palavras em inglês **ascending** e **descending**, ou seja, classificação crescente ou decrescente.

Quando não especificamos nenhum , o padrão é ascendente

Exemplo:

```
SELECT * FROM aluno ORDER BY nascimento DESC, nome ASC;
```

Verificação de caracteres

Para verificar sequência de caracteres dentro de um campo do tipo STRING (CHAR ou VARCHAR), pode-se utilizar junto com a cláusula WHERE uma condição baseada no uso do operador LIKE.

<expressão> [NOT] LIKE <valor>

Exemplos:

```
'A%' -- começa com letra A  
'_A%' -- segunda letra do nome A  
'%AN%' -- possui AN em qualquer posição
```

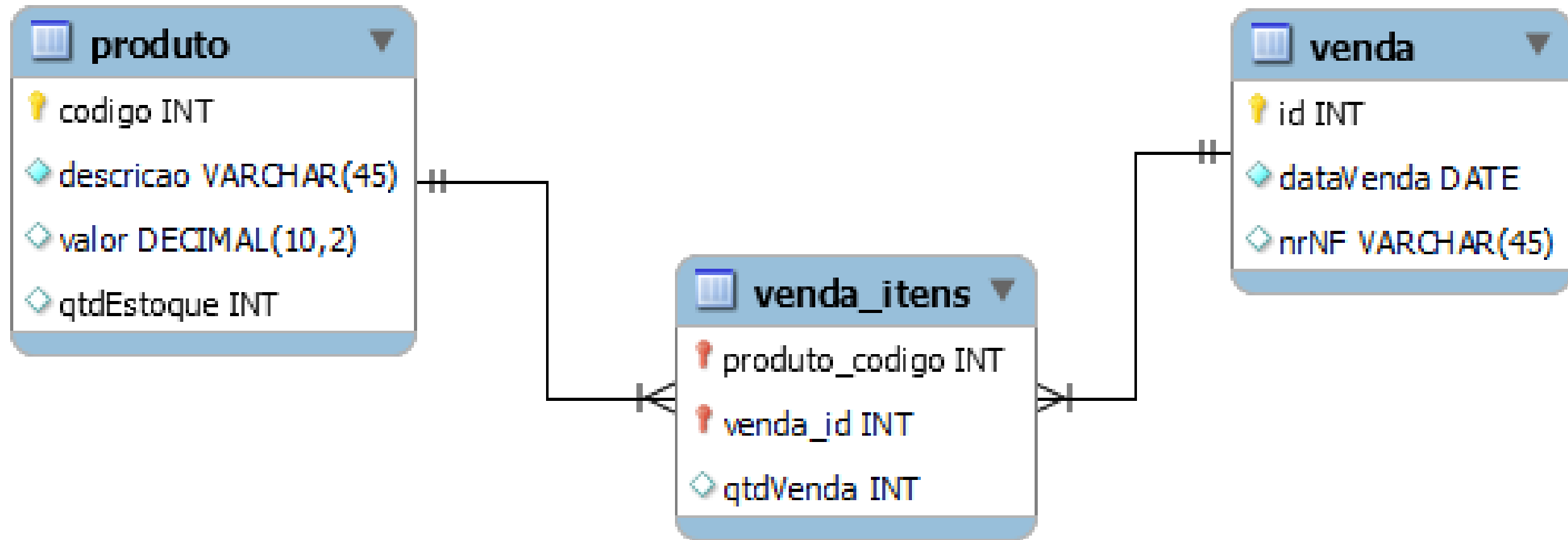
Funções Agregadas

AVG()	-- média aritmética
MAX()	-- Maior valor
MIN()	-- Menor valor
SUM()	-- Soma dos valores
COUNT()	-- Número de valores
DISTINCT	-- Contagem dos valores não vazios e únicos

A close-up, low-angle shot of a red running track. The track is composed of reddish-brown granular material. Several white lane markings are visible, including a prominent diagonal line running from the bottom left towards the top right, and several horizontal lines. A large, white, slightly irregular circle is superimposed on the right side of the image, containing the word "EXERCÍCIO" in a dark blue, sans-serif font.

EXERCÍCIO

Observe o diagrama



Resposta – Parte 1

01 - Criar a modelagem física para o diagrama apresentado. Observe, atentamente, os tipos de dados de cada campo, as chaves primárias, as chaves estrangeiras e os campos obrigatórios.

02 - Cadastrar 18 produtos (sugestão nos slides seguintes);

03 - Cadastrar 3 Vendas (sugestão nos slides seguintes);

04 - Cadastrar 10 itens de venda (sugestão nos slides seguintes);

Inserção de produtos

```
/* 02 - Cadastrar 18 produtos (sugestão no slide seguinte); */  
INSERT INTO produto (descricao, valor, qtdEstoque)  
VALUES  
( 'Camiseta Polo', 49.99, 100),  
( 'Calça Jeans', 89.99, 0),  
( 'Vestido Floral', 69.99, 60),  
( 'Moletom com Capuz', 79.99, 70),  
( 'Sapato Social', 149.99, 40),  
( 'Bolsa de Couro', 99.99, 90),  
( 'Óculos de Sol', 59.99, 110),  
( 'Jaqueta de Couro', 199.99, 30),  
( 'Saia Plissada', 39.99, 80),  
( 'Blusa de Tricô', 49.99, 70),  
( 'Calçado Infantil', 29.99, 120),  
( 'Blazer Feminino', 89.99, 50),  
( 'Calça Social', 79.99, 60),  
( 'Relógio de Pulso', 149.99, 40),  
( 'Bermuda Esportiva', 34.99, 0),  
( 'Meia Esportiva', 9.99, 150),  
( 'Chapéu de Praia', 19.99, 100),  
( 'Carteira de Couro', 39.99, 80);
```

Inserção de vendas e itens de venda

```
/* 03 - Cadastrar 3 Vendas (sugestão no slide seguinte); */
```

```
INSERT INTO venda (dataVenda, nrNF)
```

```
VALUES
```

```
('2024-05-20', '543B'),
```

```
('2024-05-20', '8567'),
```

```
('2024-05-21', '9823');
```

```
/* 04 - Cadastrar 10 itens de venda (sugestão no slide seguinte); */
```

```
INSERT INTO venda_itens (produto_codigo, venda_id, qtdVenda)
```

```
VALUES
```

```
(4, 1, 10),
```

```
(1, 1, 5),
```

```
(3, 1, 6),
```

```
(2, 2, 9),
```

```
(4, 2, 10),
```

```
(1, 2, 5),
```

```
(3, 3, 6),
```

```
(2, 3, 9),
```

```
(1, 3, 6),
```

```
(9, 3, 9);
```

Resposta – Parte 2

- 05 - Listar todos os campos de todos os produtos em ordem alfabética (crescente) de descricao;
- 06 - Listar descricao e qtdEstoque dos produtos com qtdEstoque menor do que 90;
- 07 - Alterar para 'Produto esgotado' o nome de todos os produtos com qtdEstoque menor ou igual a zero;
- 08 - Listar a dataVenda e nrNF de todas as vendas em ordem decrescente de dataVenda;
- 09 - Alterar para '2024-04-22' a dataVenda de todas as vendas;
- 10 - Listar todos os registros da tabela item, em ordem decrescente de qtdVenda.
- 11 – Liste os produtos que a descrição comece com a letra 'C'.
- 12 – Liste os produtos que contenham ' de ' na descrição;
- 13 – Liste o valor da maior quantidade em estoque;
- 14 – Liste o valor médio da quantidade de produtos em estoque;
- 15 – Liste a quantidade total vendida (soma das quantidades vendidas de todos os produtos);
- 16 – Liste as diferentes quantidades em estoque;
- 17 – Liste os diferentes valores de produto;
- 18 – Liste o valor do produto mais caro;
- 19 – Liste o valor do produto mais barato;
- 20 – Liste produtos que NÃO comecem com a letra 'C'.